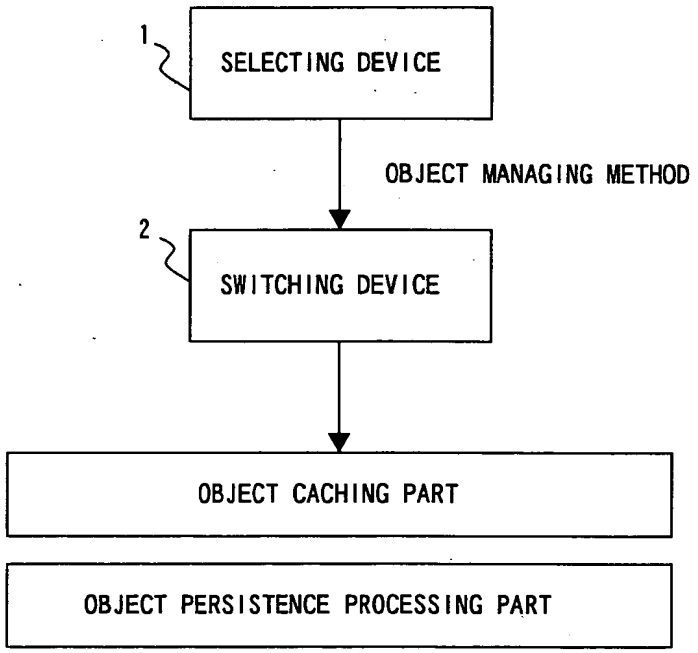
1
SELECTING DEVICE
OBJECT MANAGING METHOD
2
SWITCHING DEVICE
OBJECT CACHING PART
OBJECT PERSISTENCE PROCESSING PART

F I G. 1

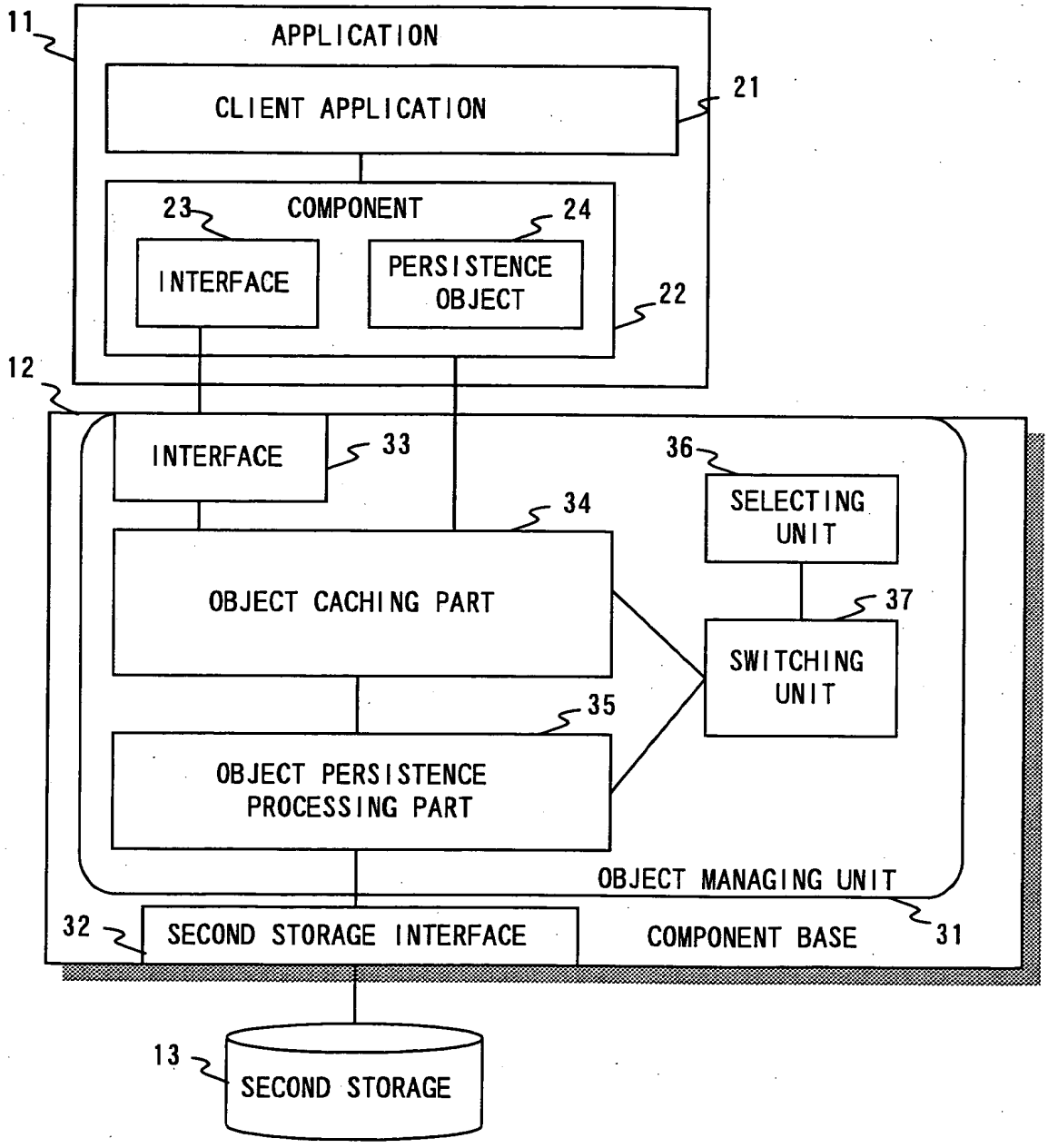
11
APPLICATION
CLIENT APPLICATION
21
23
COMPONENT
24
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
36
SELECTING UNIT
34
OBJECT CACHING PART
37
SWITCHING UNIT
35
OBJECT PERSISTENCE PROCESSING PART
OBJECT MANAGING UNIT
32
SECOND STORAGE INTERFACE
COMPONENT BASE
31
13
SECOND STORAGE

FIG. 2

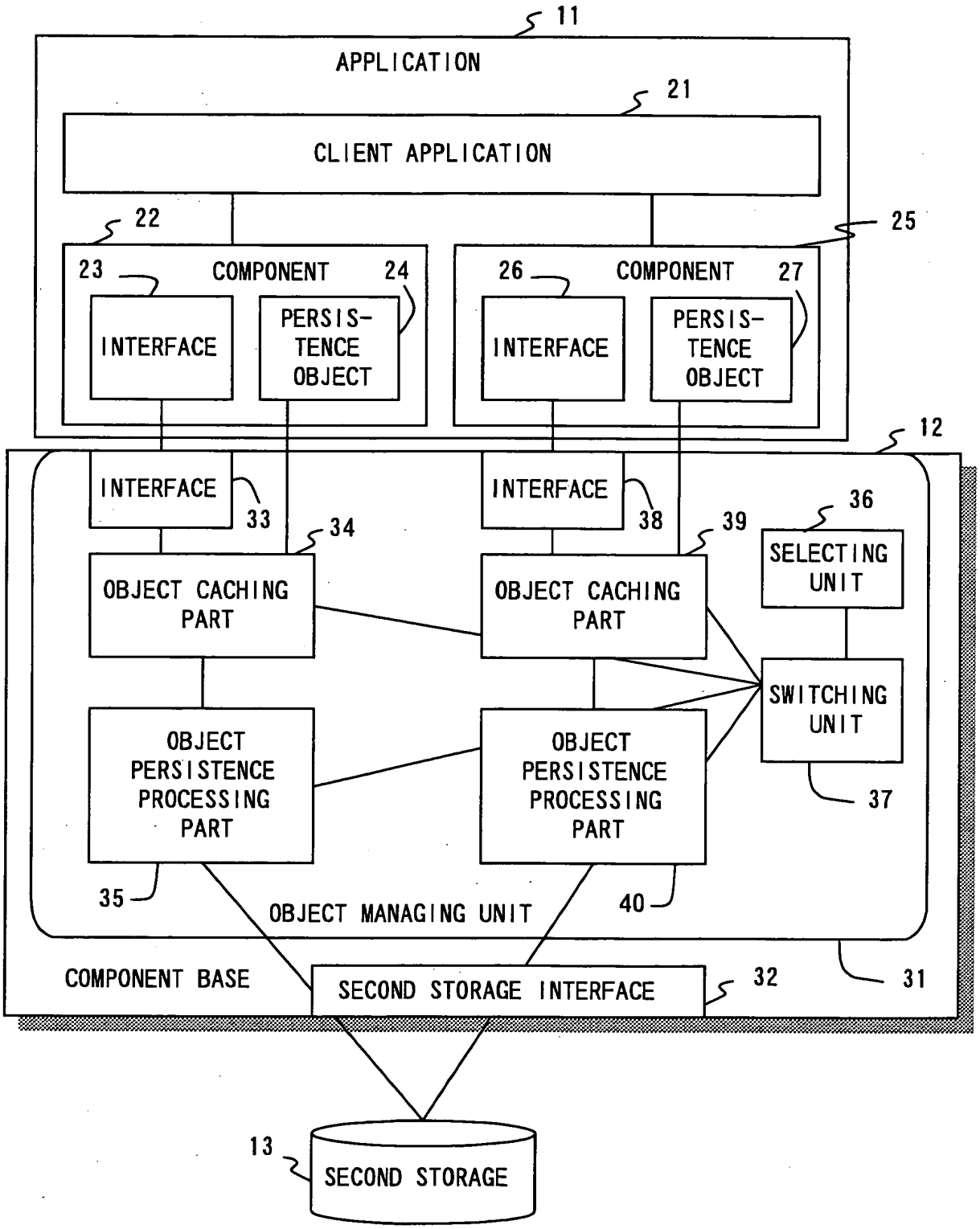
11
APPLICATION
21
CLIENT APPLICATION
22
25
23
COMPONENT
24
INTERFACE
PERSIS-
TENCE
OBJECT
26
COMPONENT
27
INTERFACE
PERSIS-
TENCE
OBJECT
12
INTERFACE
33
INTERFACE
38
36
34
39
SELECTING
UNIT
OBJECT CACHING
PART
OBJECT CACHING
PART
SWITCHING
UNIT
OBJECT
PERSISTENCE
PROCESSING
PART
OBJECT
PERSISTENCE
PROCESSING
PART
37
35
40
OBJECT MANAGING UNIT
COMPONENT BASE
SECOND STORAGE INTERFACE
32
31
13
SECOND STORAGE

FIG. 3

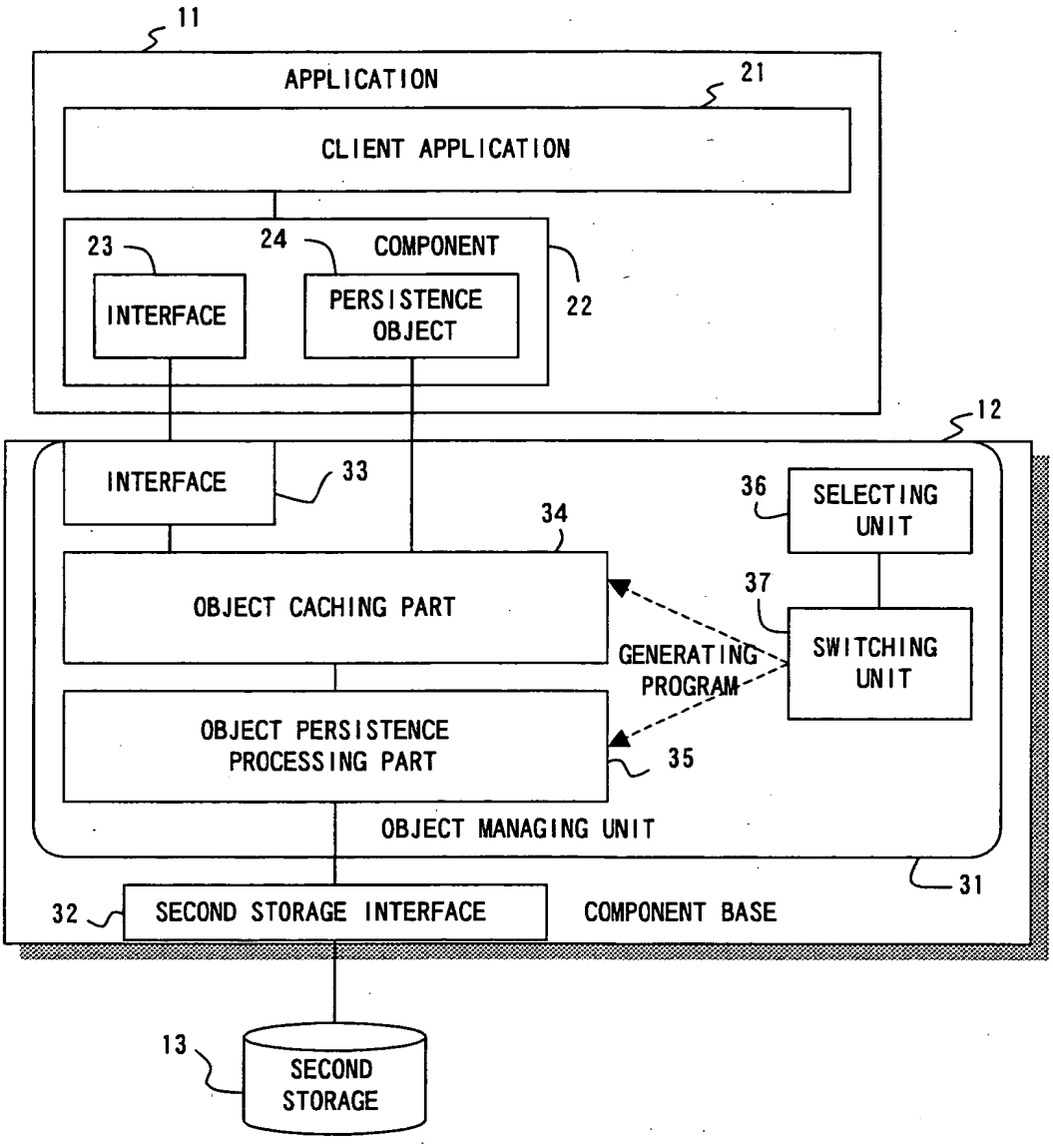
11
APPLICATION
21
CLIENT APPLICATION
23
24
COMPONENT
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
36
SELECTING UNIT
34
OBJECT CACHING PART
37
GENERATING PROGRAM
SWITCHING UNIT
OBJECT PERSISTENCE PROCESSING PART
35
OBJECT MANAGING UNIT
31
32
SECOND STORAGE INTERFACE
COMPONENT BASE
13
SECOND STORAGE

FIG. 4

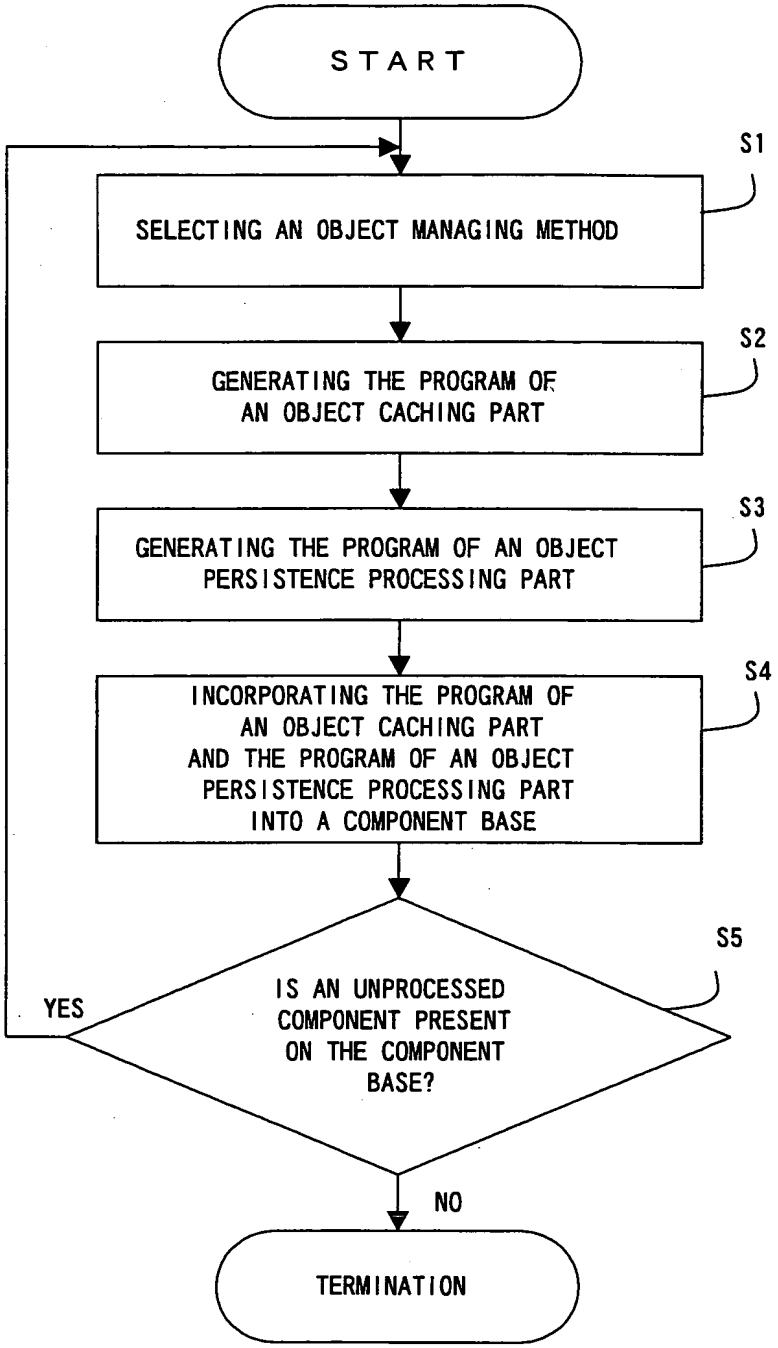

FIG. 5

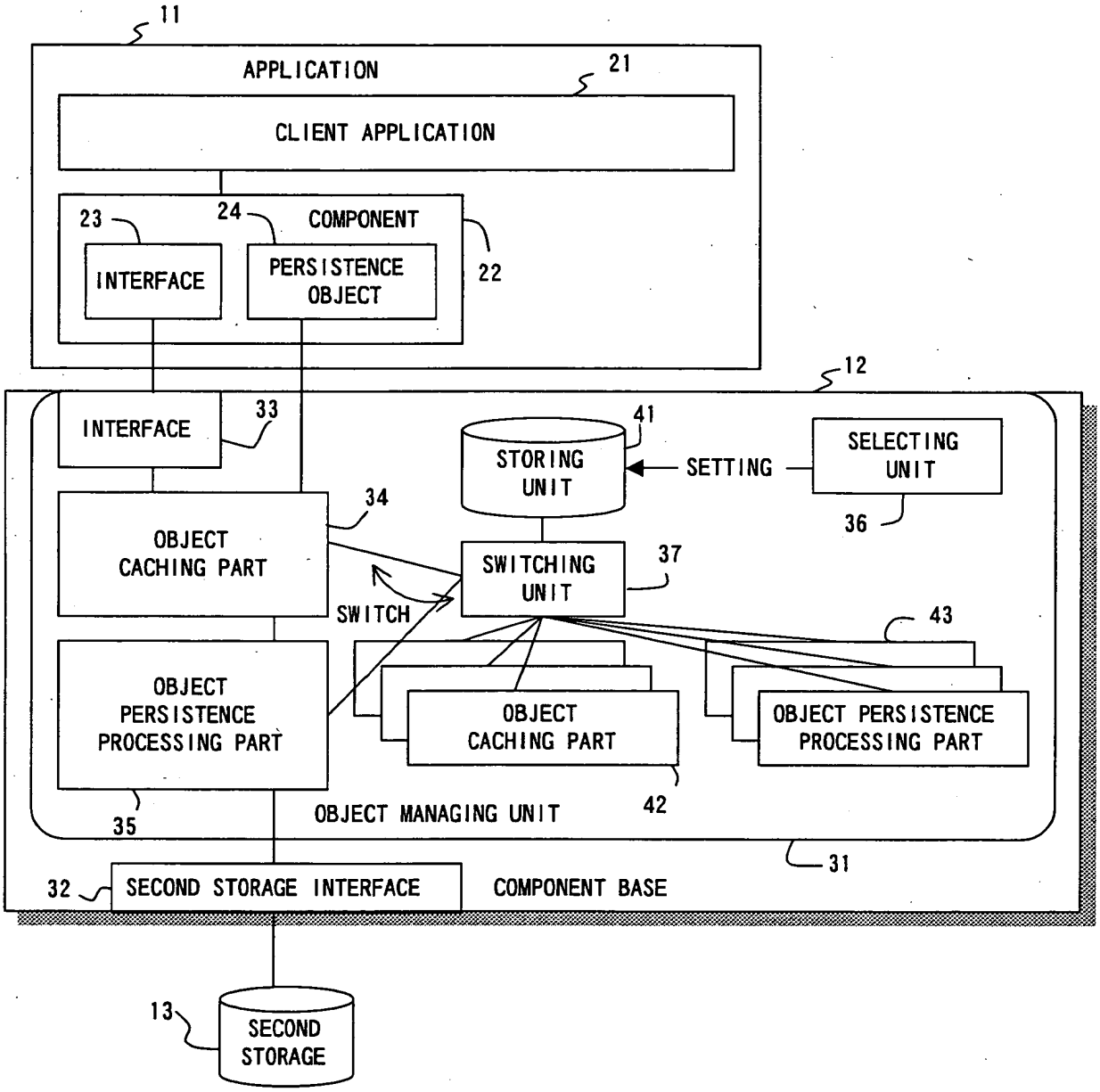
11
APPLICATION
21
CLIENT APPLICATION
23
24
COMPONENT
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
41
STORING UNIT
SETTING
SELECTING UNIT
36
34
OBJECT CACHING PART
SWITCHING UNIT
37
SWITCH
43
OBJECT PERSISTENCE PROCESSING PART
OBJECT CACHING PART
OBJECT PERSISTENCE PROCESSING PART
35
OBJECT MANAGING UNIT
42
31
32
SECOND STORAGE INTERFACE
COMPONENT BASE
13
SECOND STORAGE

FIG. 6

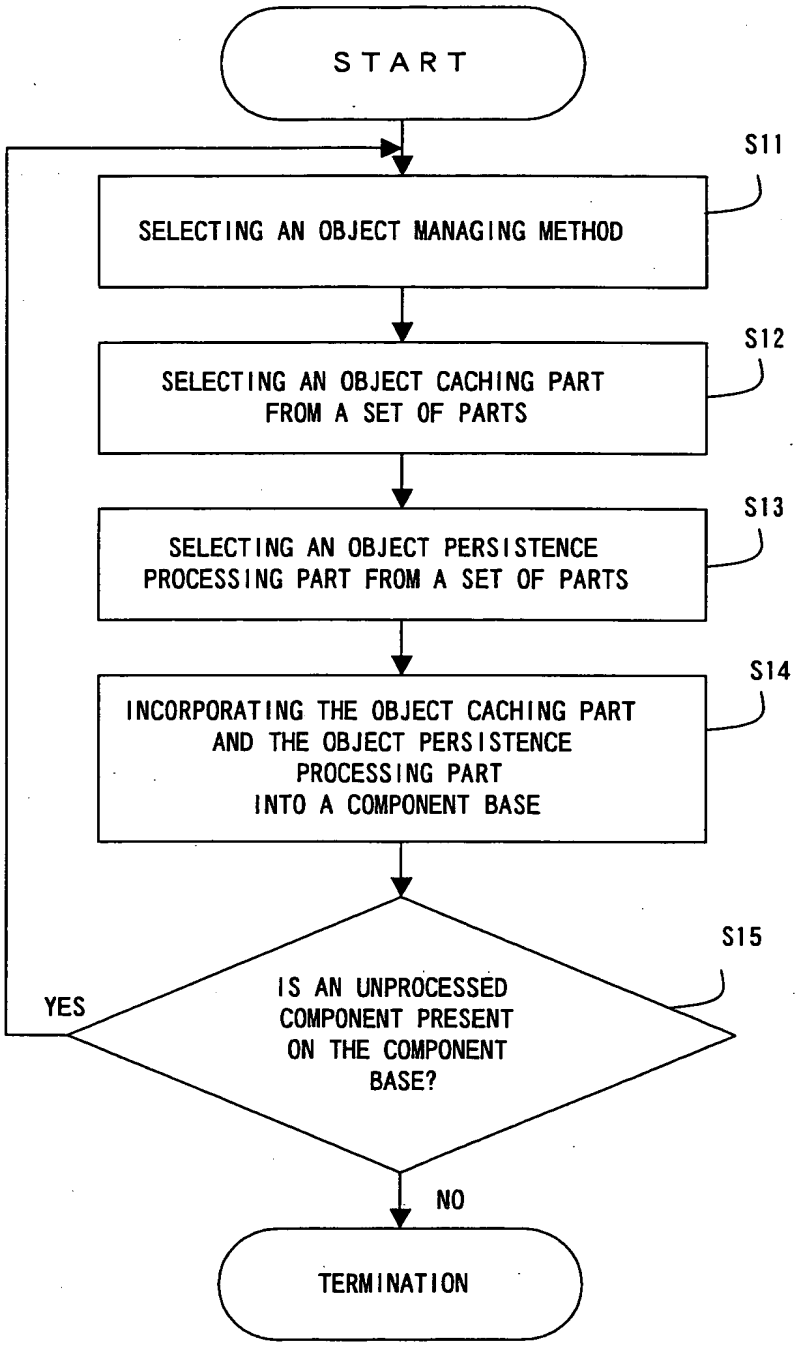
START
S11
SELECTING AN OBJECT MANAGING METHOD
S12
SELECTING AN OBJECT CACHING PART FROM A SET OF PARTS
S13
SELECTING AN OBJECT PERSISTENCE PROCESSING PART FROM A SET OF PARTS
S14
INCORPORATING THE OBJECT CACHING PART AND THE OBJECT PERSISTENCE PROCESSING PART INTO A COMPONENT BASE
S15
IS AN UNPROCESSED COMPONENT PRESENT ON THE COMPONENT BASE?
YES
NO
TERMINATION

FIG. 7

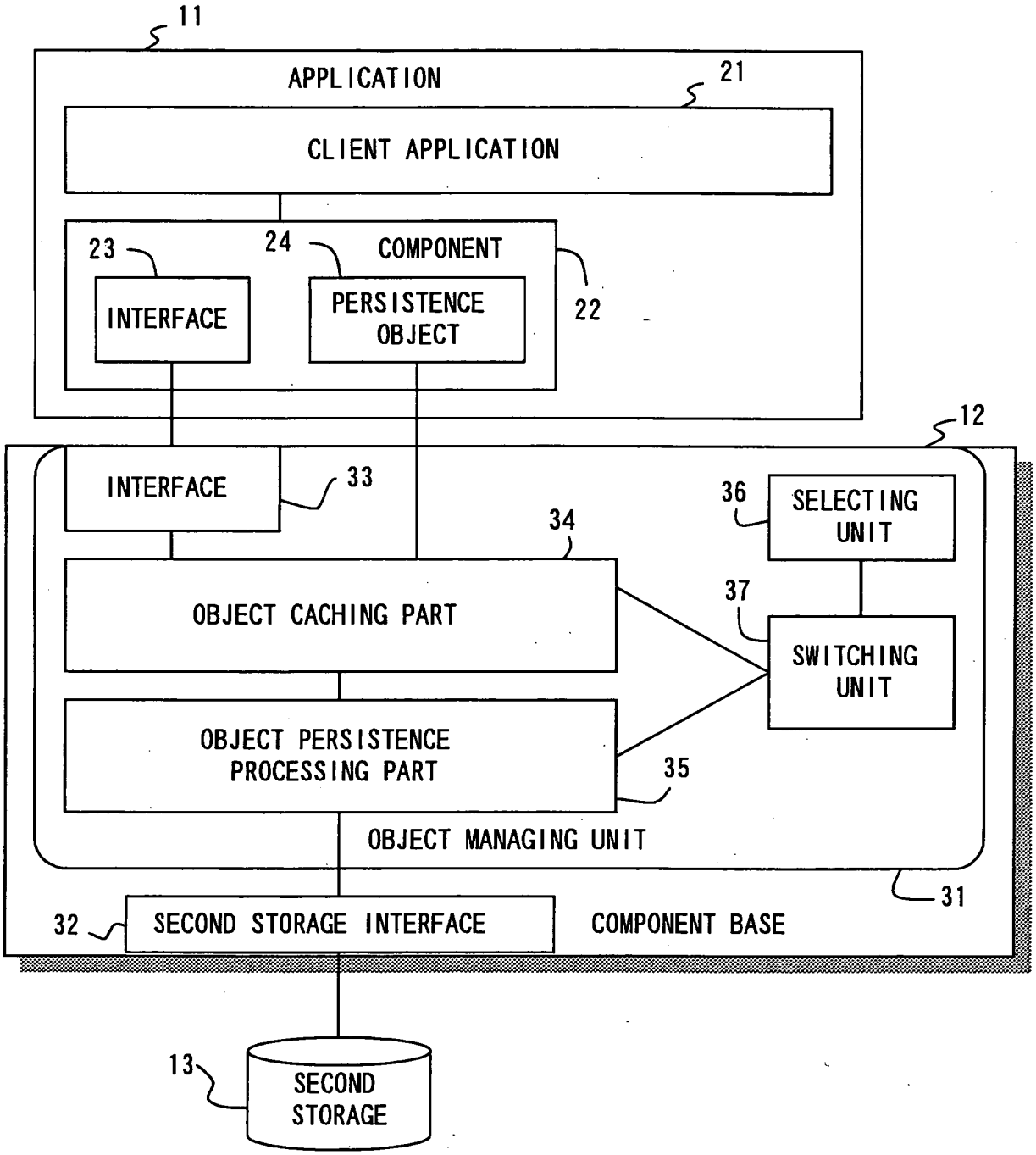
11
APPLICATION
21
CLIENT APPLICATION
23
24
COMPONENT
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
36
SELECTING UNIT
34
OBJECT CACHING PART
37
SWITCHING UNIT
OBJECT PERSISTENCE PROCESSING PART
35
OBJECT MANAGING UNIT
31
32
SECOND STORAGE INTERFACE
COMPONENT BASE
13
SECOND STORAGE

FIG. 8

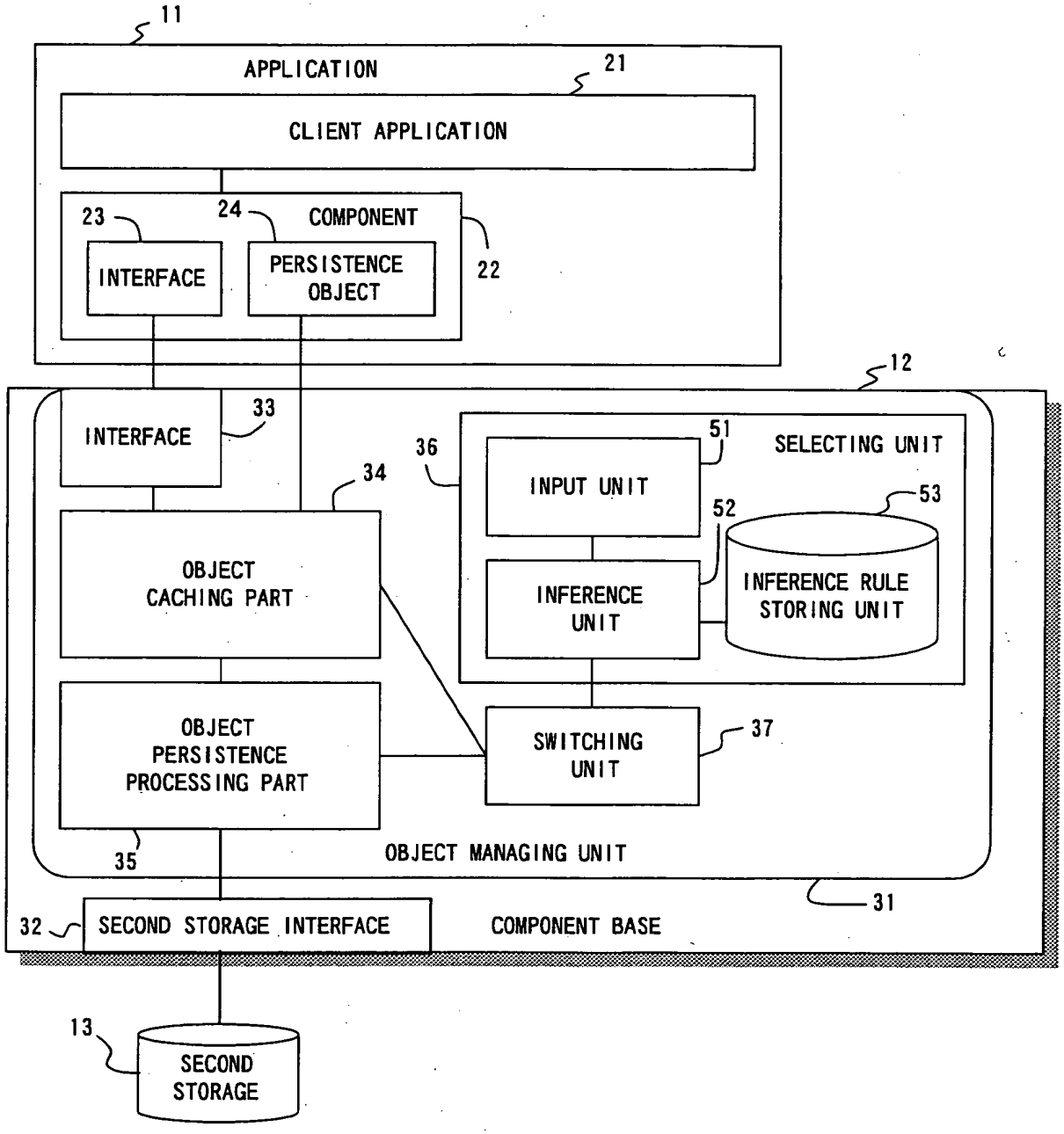
11
APPLICATION
21
CLIENT APPLICATION
23
24
COMPONENT
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
36
51
SELECTING UNIT
34
INPUT UNIT
52
53
OBJECT CACHING PART
INFERENCE UNIT
INFERENCE RULE STORING UNIT
OBJECT PERSISTENCE PROCESSING PART
SWITCHING UNIT
37
35
OBJECT MANAGING UNIT
31
32
SECOND STORAGE INTERFACE
COMPONENT BASE
13
SECOND STORAGE

FIG. 9

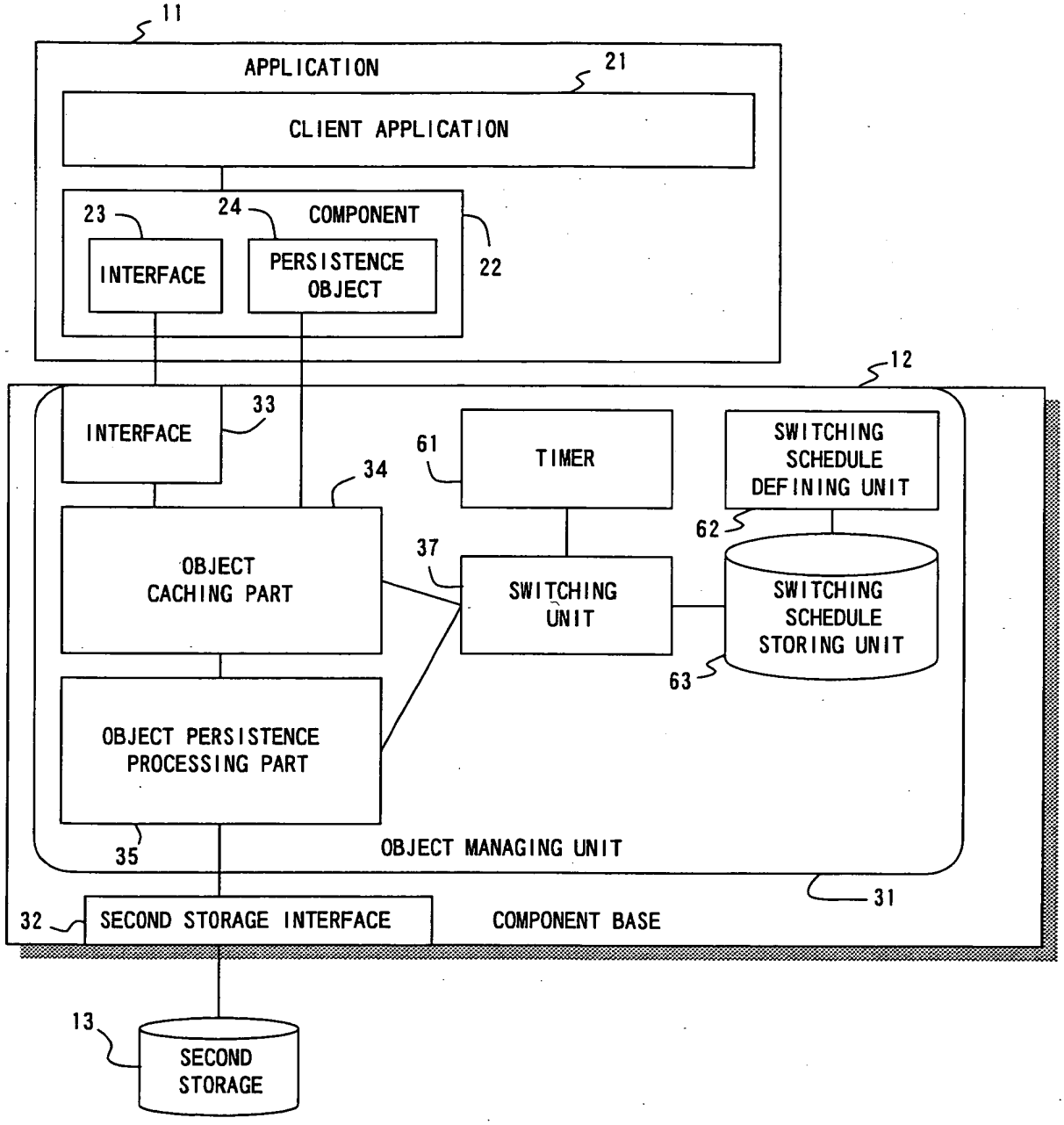
11
APPLICATION
21
CLIENT APPLICATION
23
24
COMPONENT
INTERFACE
PERSISTENCE OBJECT
22
12
INTERFACE
33
61
TIMER
SWITCHING SCHEDULE DEFINING UNIT
34
62
OBJECT CACHING PART
37
SWITCHING UNIT
SWITCHING SCHEDULE STORING UNIT
63
OBJECT PERSISTENCE PROCESSING PART
35
OBJECT MANAGING UNIT
31
32
SECOND STORAGE INTERFACE
COMPONENT BASE
13
SECOND STORAGE

FIG. 10

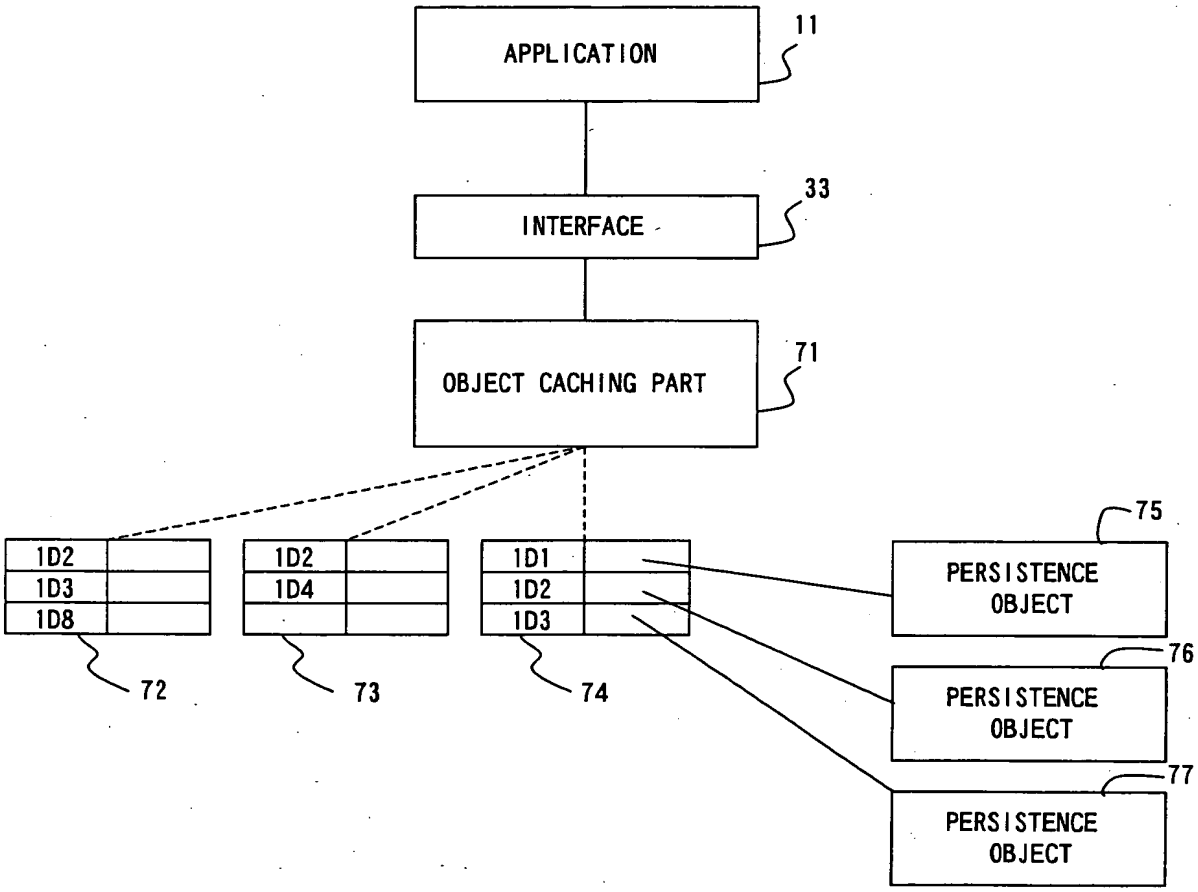
APPLICATION
11
INTERFACE
33
OBJECT CACHING PART
71
1D2
1D3
1D8
72
1D2
1D4
73
1D1
1D2
1D3
74
75
PERSISTENCE OBJECT
76
PERSISTENCE OBJECT
77
PERSISTENCE OBJECT

FIG. 11

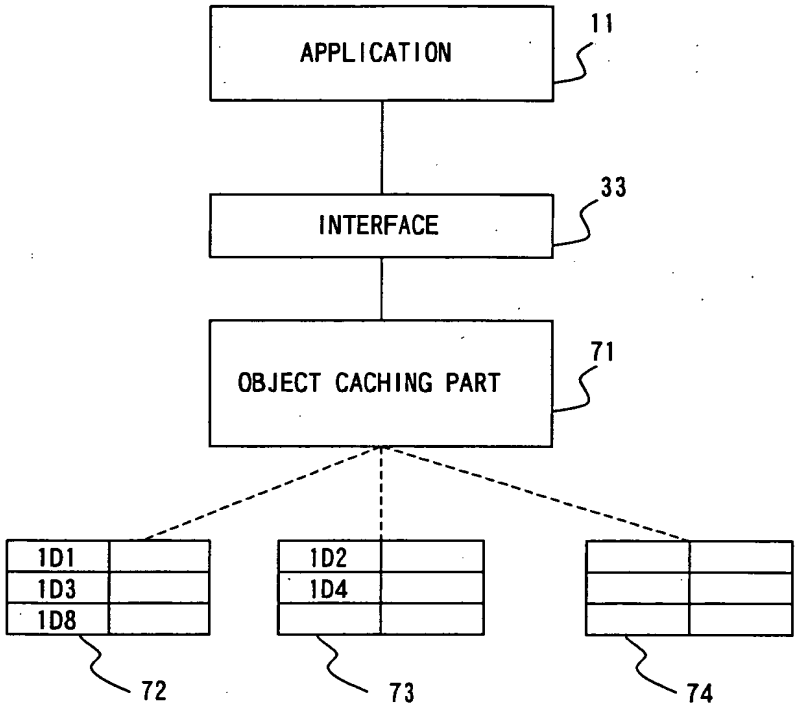
APPLICATION
11
INTERFACE
33
OBJECT CACHING PART
71
1D1
1D3
1D8
72
1D2
1D4
73
74

FIG. 12

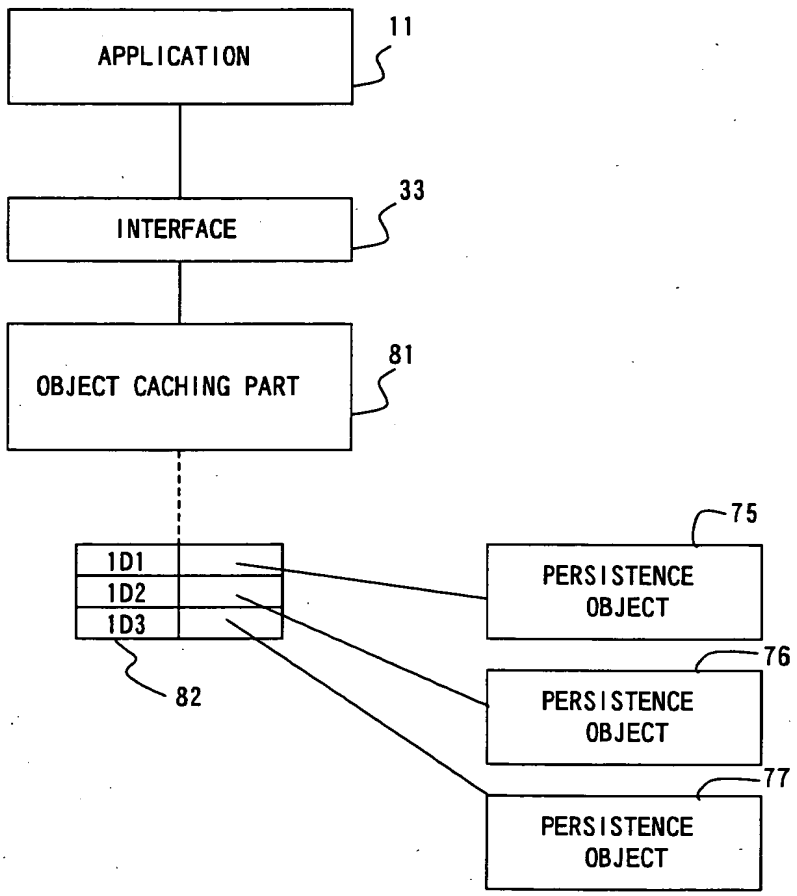
APPLICATION
11
INTERFACE
33
OBJECT CACHING PART
81
1D1
1D2
1D3
82
75
PERSISTENCE OBJECT
76
PERSISTENCE OBJECT
77
PERSISTENCE OBJECT

FIG. 13

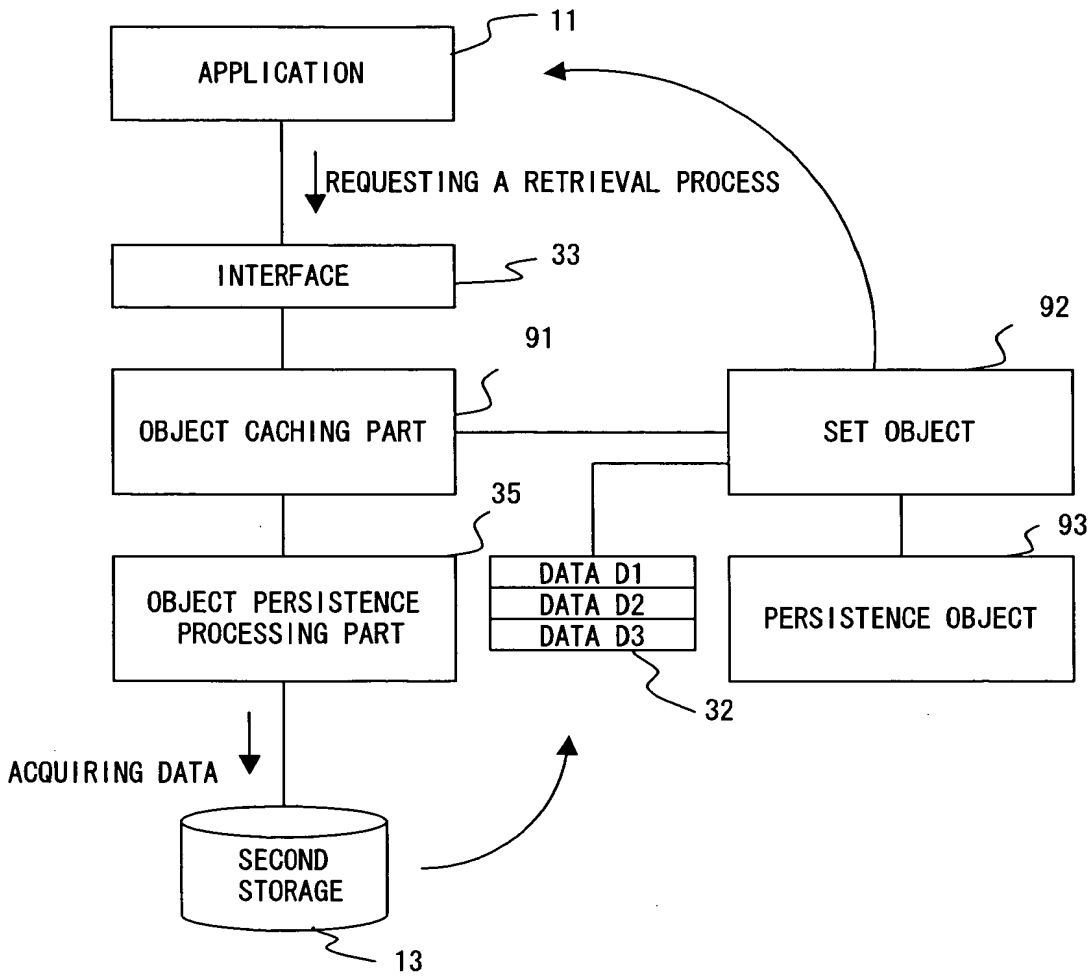
APPLICATION
11
REQUESTING A RETRIEVAL PROCESS
INTERFACE
33
OBJECT CACHING PART
91
SET OBJECT
92
OBJECT PERSISTENCE PROCESSING PART
35
DATA D1
DATA D2
DATA D3
32
PERSISTENCE OBJECT
93
ACQUIRING DATA
SECOND STORAGE
13

FIG. 14

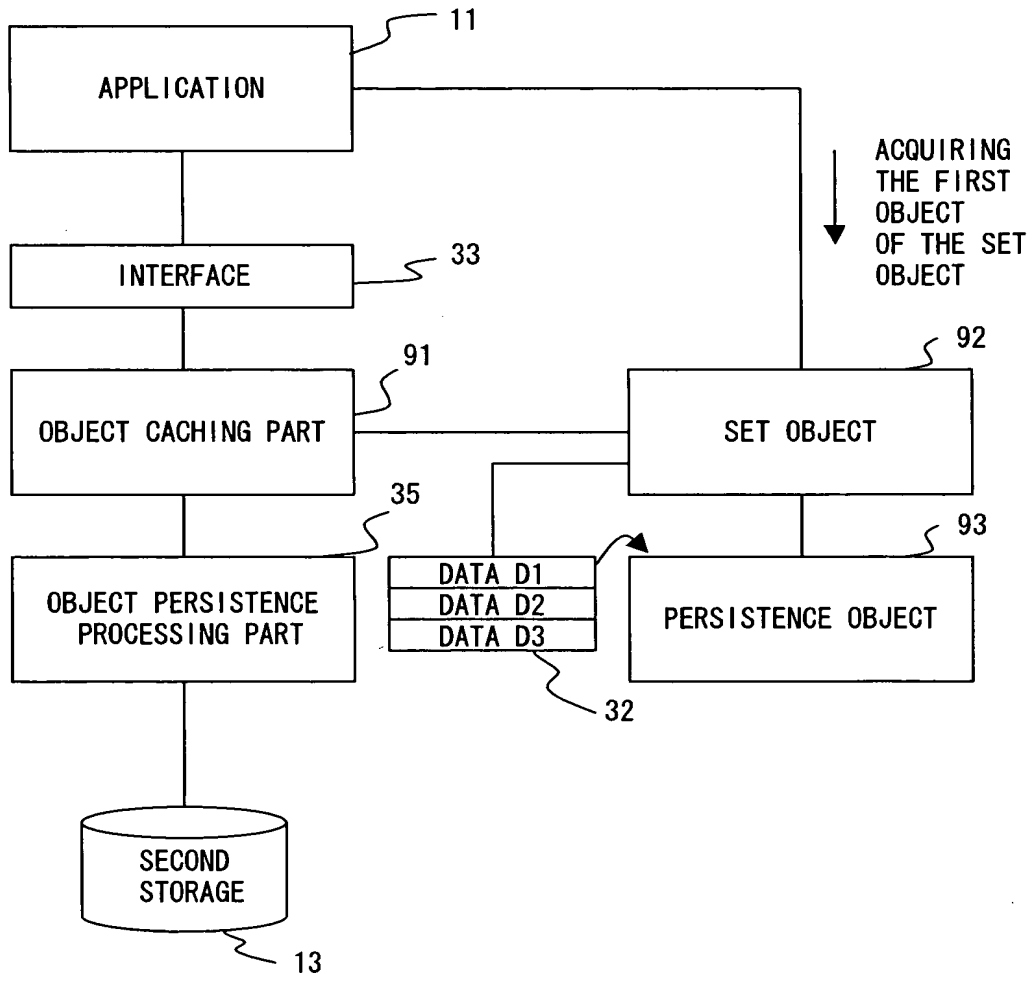
11
APPLICATION
ACQUIRING THE FIRST OBJECT OF THE SET OBJECT
INTERFACE
33
91
92
OBJECT CACHING PART
SET OBJECT
35
93
OBJECT PERSISTENCE PROCESSING PART
DATA D1
DATA D2
DATA D3
PERSISTENCE OBJECT
32
SECOND STORAGE
13

FIG. 15

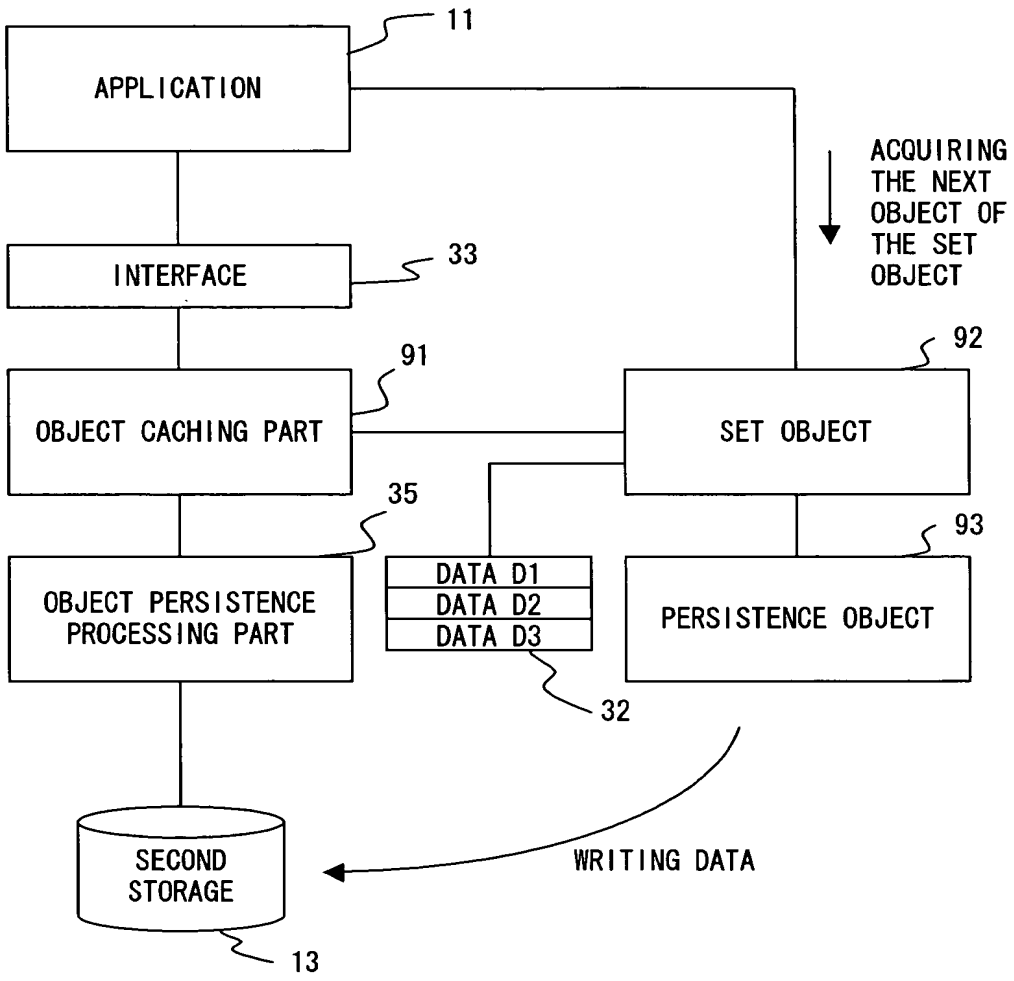

FIG. 16

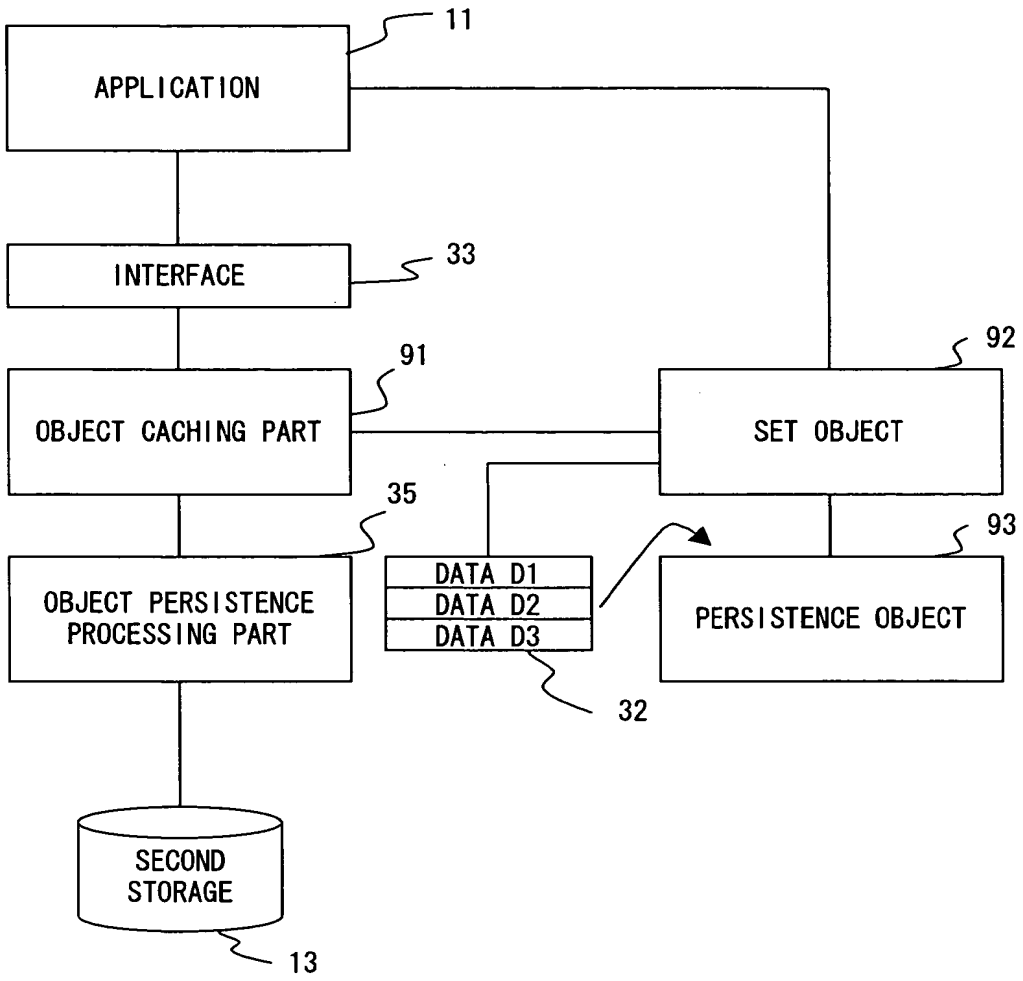
11
APPLICATION
INTERFACE
33
91
OBJECT CACHING PART
92
SET OBJECT
35
93
OBJECT PERSISTENCE PROCESSING PART
DATA D1
DATA D2
DATA D3
32
PERSISTENCE OBJECT
SECOND STORAGE
13

FIG. 17

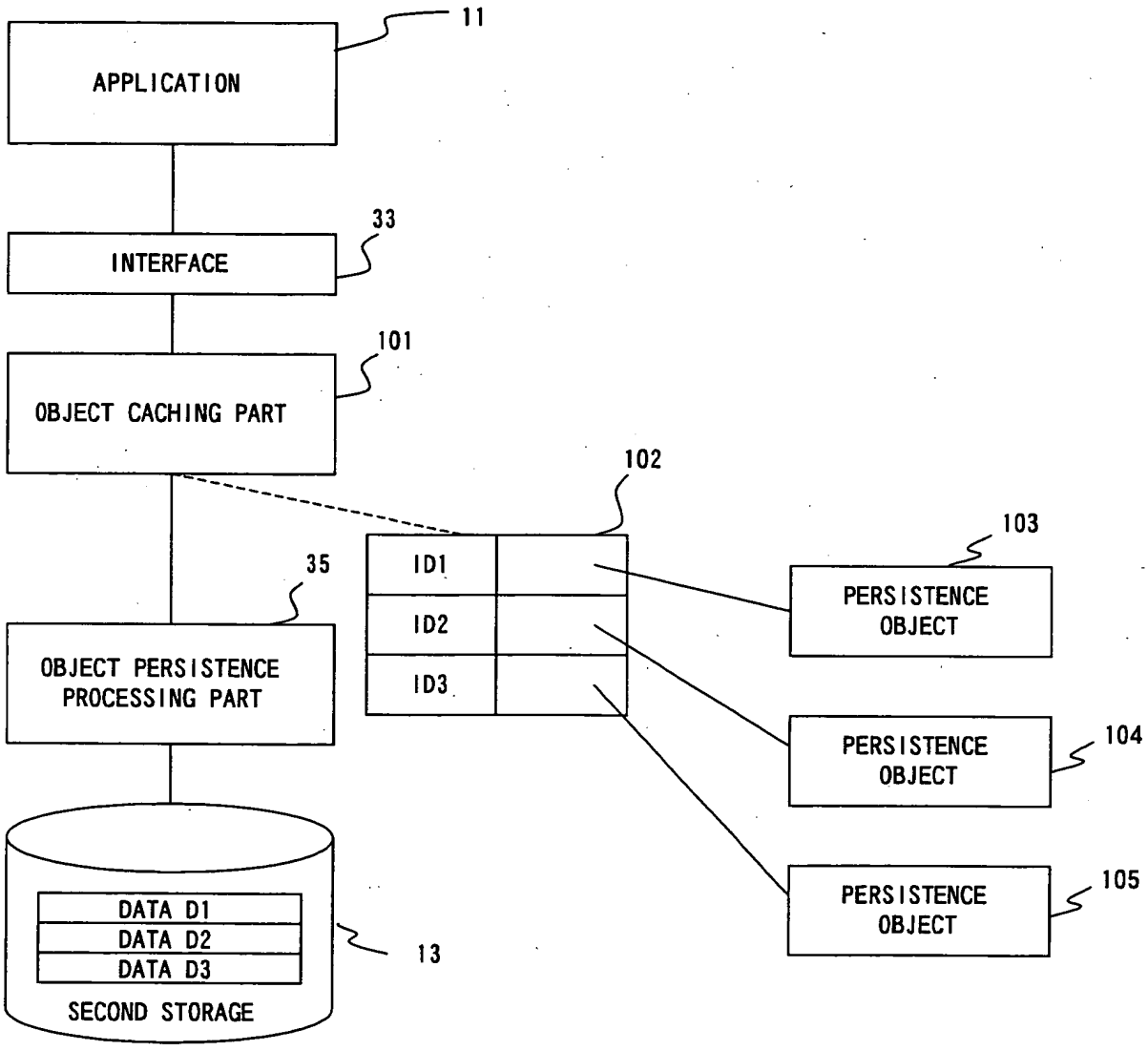
11
APPLICATION
33
INTERFACE
101
OBJECT CACHING PART
102
ID1
ID2
ID3
103
PERSISTENCE OBJECT
104
PERSISTENCE OBJECT
105
PERSISTENCE OBJECT
35
OBJECT PERSISTENCE PROCESSING PART
DATA D1
DATA D2
DATA D3
13
SECOND STORAGE

FIG. 18

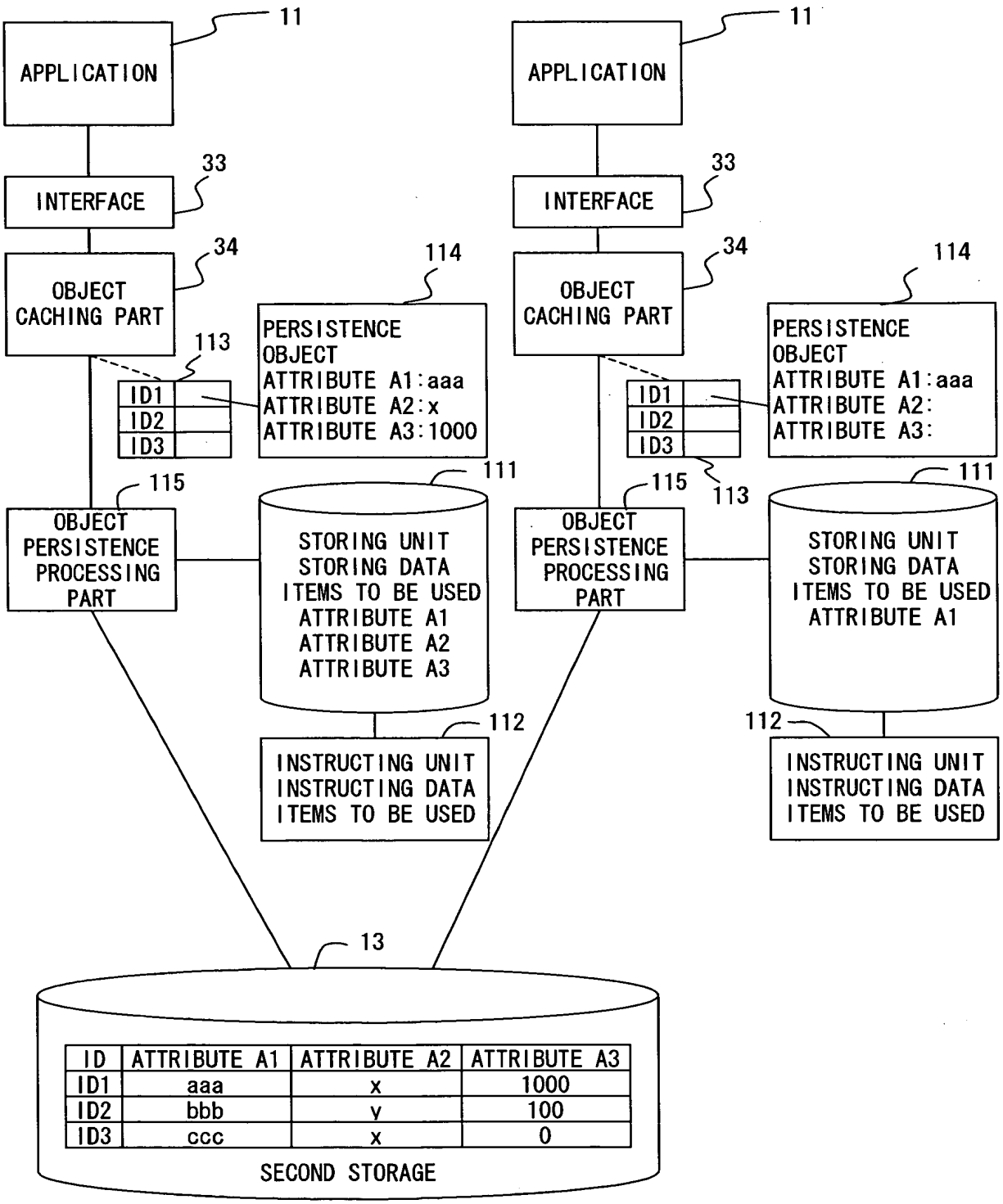
APPLICATION
11
INTERFACE
33
OBJECT CACHING PART
34
113
ID1
ID2
ID3
PERSISTENCE OBJECT
ATTRIBUTE A1:aaa
ATTRIBUTE A2:x
ATTRIBUTE A3:1000
114
OBJECT PERSISTENCE PROCESSING PART
115
STORING UNIT
STORING DATA ITEMS TO BE USED
ATTRIBUTE A1
ATTRIBUTE A2
ATTRIBUTE A3
111
INSTRUCTING UNIT
INSTRUCTING DATA ITEMS TO BE USED
112
APPLICATION
11
INTERFACE
33
OBJECT CACHING PART
34
113
ID1
ID2
ID3
PERSISTENCE OBJECT
ATTRIBUTE A1:aaa
ATTRIBUTE A2:
ATTRIBUTE A3:
114
OBJECT PERSISTENCE PROCESSING PART
115
STORING UNIT
STORING DATA ITEMS TO BE USED
ATTRIBUTE A1
111
INSTRUCTING UNIT
INSTRUCTING DATA ITEMS TO BE USED
112
13
ID
ATTRIBUTE A1
ATTRIBUTE A2
ATTRIBUTE A3
ID1
aaa
x
1000
ID2
bbb
y
100
ID3
ccc
x
0
SECOND STORAGE

FIG. 19

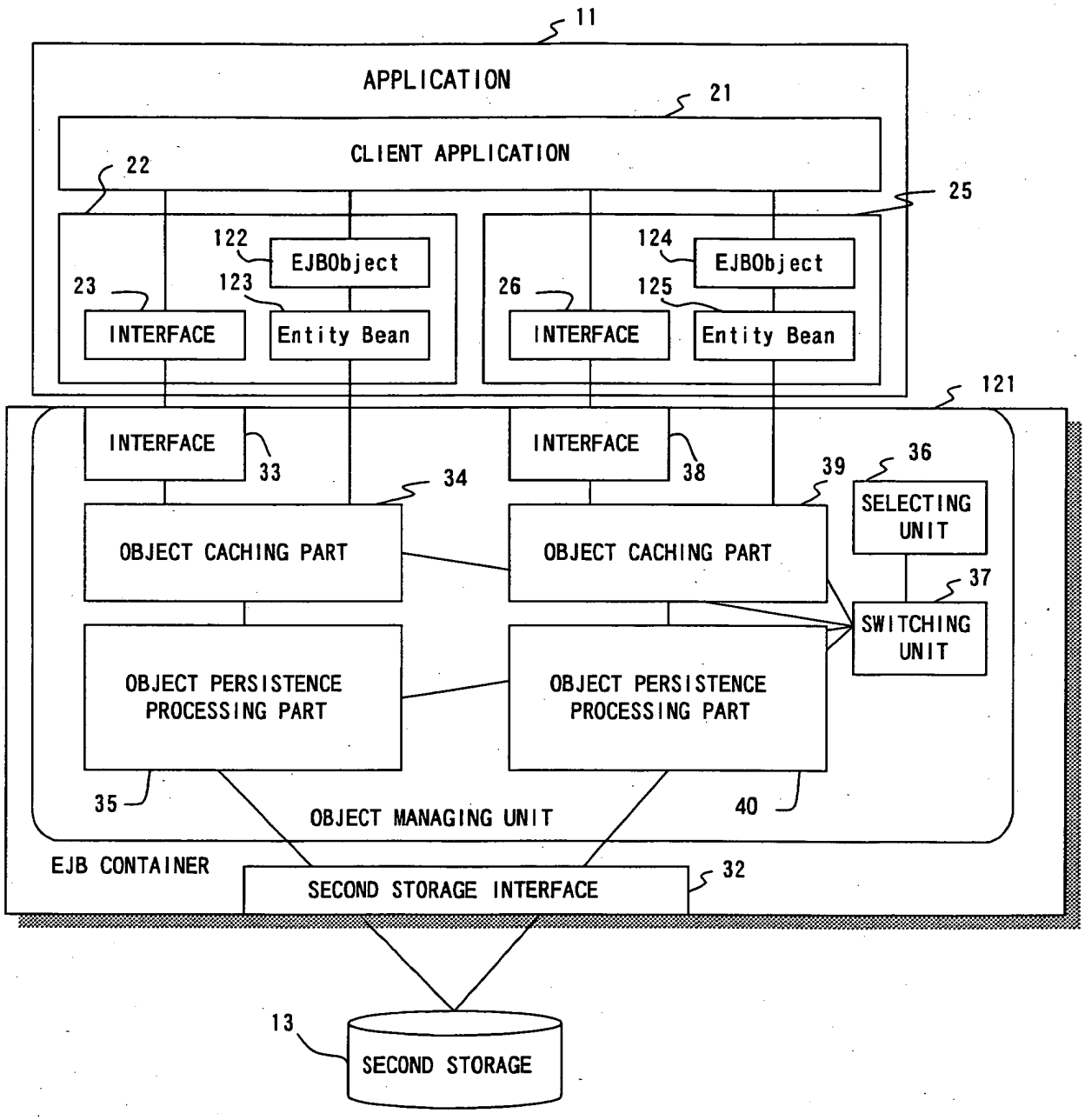
11
APPLICATION
21
CLIENT APPLICATION
22
25
122
EJBObject
124
EJBObject
23
123
26
125
INTERFACE
Entity Bean
INTERFACE
Entity Bean
121
INTERFACE
33
INTERFACE
38
34
39
36
SELECTING UNIT
OBJECT CACHING PART
OBJECT CACHING PART
37
SWITCHING UNIT
OBJECT PERSISTENCE PROCESSING PART
OBJECT PERSISTENCE PROCESSING PART
35
40
OBJECT MANAGING UNIT
EJB CONTAINER
SECOND STORAGE INTERFACE
32
13
SECOND STORAGE

FIG. 20

```
public class $$Bean$$Cache
{
    Hashtable cacheTableHash;
    $$Bean$$Persistence objPersistence;

    $$Bean$$ findByPrimaryKey($$PrimaryKey$$ pk)
    {
        String transactionID = getTransactionID();
        Hashtable cacheTable = (Hashtable)cacheTableHash.get(transactionID);
        $$Bean$$ bean = cacheTable.get(pk);

        if (bean != null) return bean;

        bean = objPersistence.findBean(pk);
        cacheTable.put(pk,bean);
        return bean;
    }
    ....

}
```

FIG. 21

```
public class $$Bean$$Persistence
{
    DataSource dataSource;

    $$Bean$$ findBean($$PrimaryKey$$ pk)
    {
        $$Bean$$ bean = null;
        Connection connection = getConnection();
        String sql = "SELECT $$BeanFieldColumns$$ FROM $$Table$$ WHERE
            $$PKFieldColumns$$ = ?";
        Statement statement = connection.preparedStatement(sql);

        //$$ foreach $$PKFields$$
        statement.set$$Type$$($$Count$$,pk.$$Name$$);
        //$$ end foreach

        ResultSet rs = statement.executeQuery();
        while(rs.next())
        {
            bean = new $$Bean$$();
            //$$ foreach $$BeanFields$$
            bean.$$Name$$ = rs.get$$Type$$($$Count$$);
            //$$ end foreach

        }
        return bean;
    }
    ....
```

FIG. 22

```
public class OrderBeanCache
{
    Hashtable cacheTableHash;
    OrderBeanPersistence objPersistence;

    OrderBean findByPrimaryKey(OrderBeanPrimaryKey pk)
    {
        String transactionID = getTransactionID();
        Hashtable cacheTable = (Hashtable)cacheTableHash.get(transactionID);
        OrderBean bean = cacheTable.get(pk);

        if (bean != null) return bean;

        bean = objPersistence.findBean(pk);
        cacheTable.put(pk,bean);
        return bean;
    }
    ....
}
```

FIG. 23

```
public class OrderBeanPersistence
{
    DataSource dataSource;

    OrderBean findBean(OrderBeanPrimaryKey pk)
    {
        OrderBean bean = null;
        Connection connection = getConnection();
        String sql = "SELECT ID,PRODUCT,QUANTITY FROM ORDERTABLE
WHERE ID = ?";
        Statement statement = connection.preparedStatement(sql);

        statement.setString(1,pk.id);
        statement.setString(2,pk.product);
        statement.setInt(3,pk.quantity);

        ResultSet rs = statement.executeQuery();
        while(rs.next())
        {
            bean = new OrderBean();
            bean.id = rs.getString(1);
            bean.product = rs.getString(2);
            bean.quantity = rs.getInt(3);

        }
        return bean;
    }
....

}
```

FIG. 24

```
public class ObjectCacheOption1 extends ObjectCache
{
    Hashtable cacheTableHash;
    ObjectPersistence objPersistence;

    Object findByPrimaryKey(Object pk, BeanDef beanDef)
    {
        String transactionID = getTransactionID();
        Hashtable cacheTable = (Hashtable)cacheTableHash.get(transactionID);
        Object bean = cacheTable.get(pk);

        if (bean != null) return bean;

        bean = objPersistence.findBean(pk,beanDef,"findByPrimaryKey");
        cacheTable.put(pk,bean);
        return bean;
    }
    ....
}
```

FIG. 25

```
public class ObjectPersistenceOption1 extends ObjectPersistence
{
    DataSource dataSource;

    Object findBean(Object pk, BeanDef beanDef, String finderName)
    {
        Object bean = null;
        Connection connection = getConnection();
        String sql = beanDef.getSQL(finderName);
        Statement statement = connection.preparedStatement(sql);

        Enumeration fields = beanDef.getFields();
        while (fields.hasMoreElements())
        {
            FieldDef field = (FieldDef)fields.nextElement();
            setValue(statement,field);
            }

            ResultSet rs = statement.executeQuery();
            while(rs.next())
            {
                bean = beanDef.newInstance();
                fields = beanDef.getFields();
                while (fields.hasMoreElements())
                {
                    FieldDef field = (FieldDef)fields.nextElement();
                    setValue(bean,rs,field);
                }
            }
        }
        return bean;
    }
    ...

    void setValue(Object bean, ResultSet, FieldDef field)
    {
        Field f = field.getField();
        switch (field.fieldType)
        {
            case FT_INT:
                f.setInt(bean,rs.getInt(field.getColumn()));
                break;
            case FT_LONG:
                f.setLong(bean,rs.getLong(field.getColumn()));
                break;
            ...
        }
    }
    ...
}
```

FIG. 26

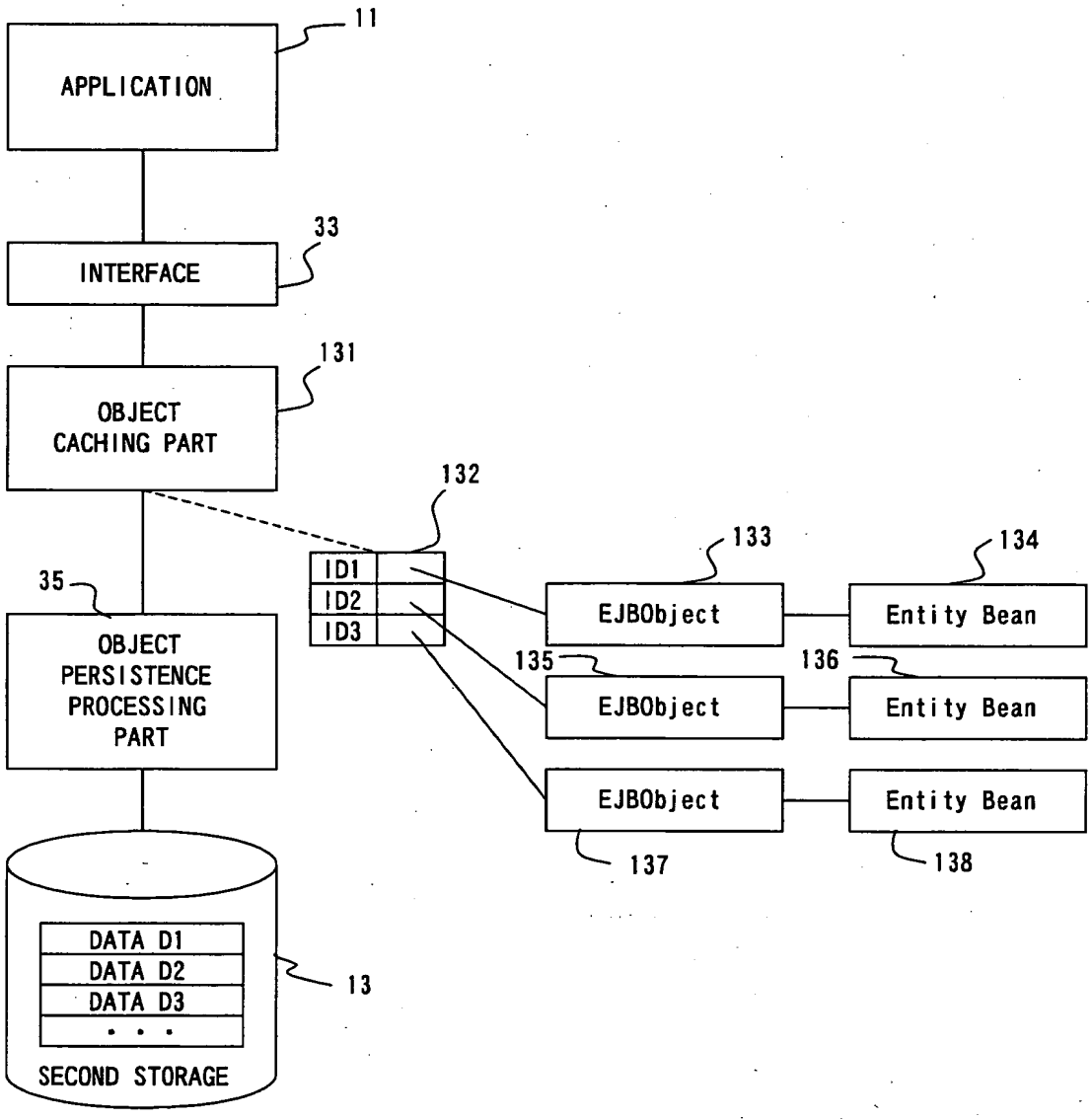

FIG. 27

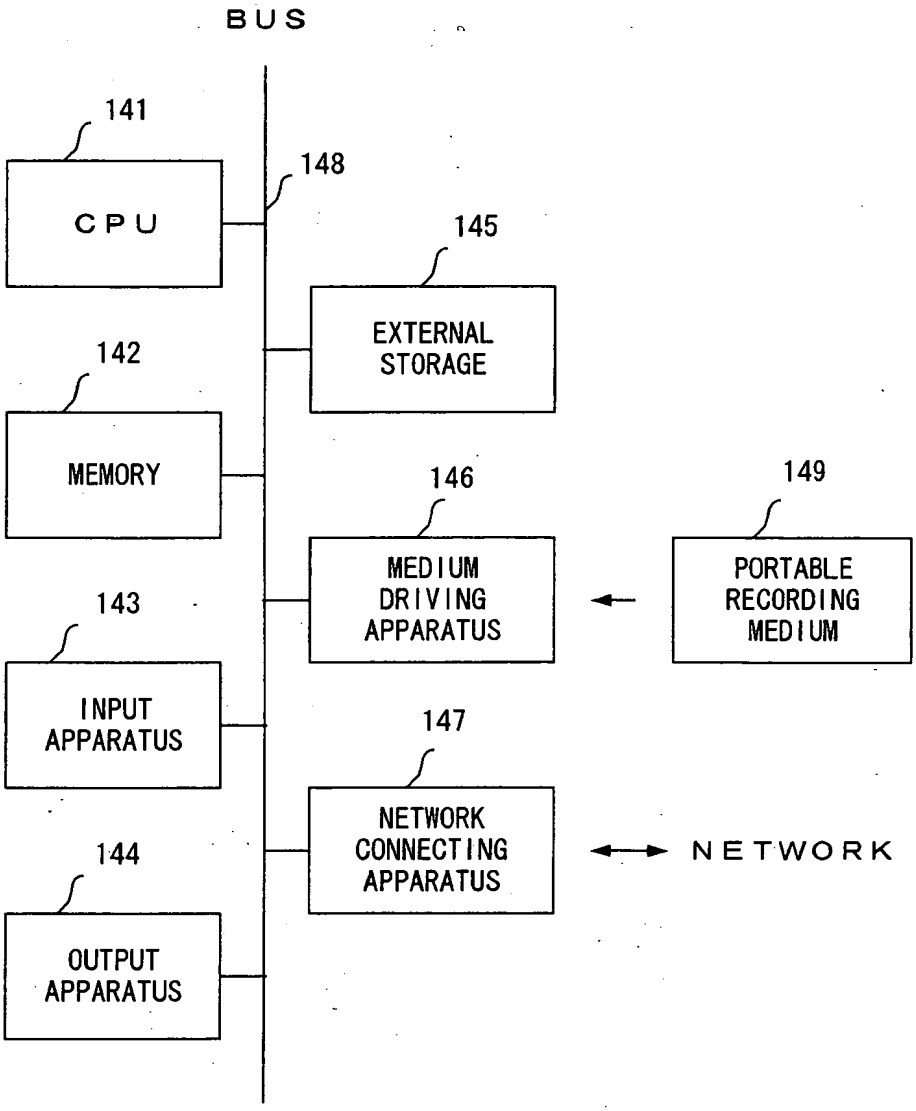
BUS
141
CPU
148
145
EXTERNAL STORAGE
142
MEMORY
146
MEDIUM DRIVING APPARATUS
149
PORTABLE RECORDING MEDIUM
143
INPUT APPARATUS
147
NETWORK CONNECTING APPARATUS
NETWORK
144
OUTPUT APPARATUS

FIG. 28

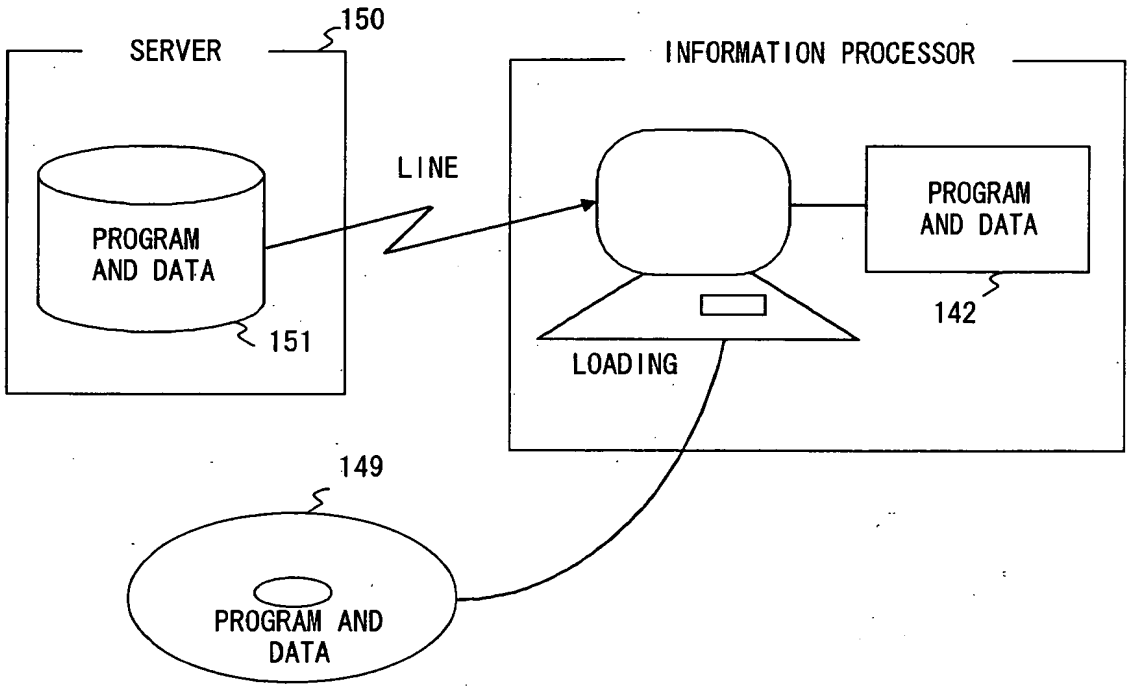
150
SERVER
INFORMATION PROCESSOR
LINE
PROGRAM
AND DATA
151
PROGRAM
AND DATA
142
LOADING
149
PROGRAM AND
DATA

FIG. 29